

L2_2.1 Datenbankmodell mit zwei Tabellen – Information

Ein Entity-Relationship-Diagramm (ERD) stellt einen Standard für die Modellierung von logischen Datenbankbeziehungen dar. Es bildet die Datenstrukturen ab und dient zum einen in der konzeptionellen Phase der Anwendungsentwicklung der Verständigung zwischen Anwendern und Entwicklern. Zum anderen ist das ER-Diagramm in der Implementierungsphase die Grundlage für das Design der Datenbank.

Bei diesen Diagrammen wird anhand einer begrenzten Anzahl von Symbolen wie Rechtecken, Rauten, Ovalen und Verbindungslinien die Vernetzung zwischen **Entitätstypen**, **Beziehungen** und Attributen dargestellt.

Zur besseren Übersicht soll im Folgenden auf die Festlegung und Darstellung der Attribute verzichtet werden. Dieser Schritt der Modellierung erfolgt im Zusammenhang mit der Überführung des ERD in ein Relationenmodell.

Entität – Entitätstyp

Zu Beginn des Datenmodellierungsprozesses müssen alle Objekte des zu analysierenden Projektes identifiziert werden. Unter einem Objekt ist dabei nicht nur ein konkreter Gegenstand (Fahrrad, Haus, etc.) zu verstehen. Vielmehr sind auch Lebewesen (Mensch, Tier, Pflanze) und abstrakte Dinge wie Rollen von Personen (Kunde, Mitarbeiter etc.) oder Ereignisse (Vermietung, Bestellung etc.) als Objekte zu verstehen.

In der Datenbanktheorie werden solche Objekte als **Entitäten** bezeichnet. Sie stellen eindeutig bestimmbare, von anderen unterscheidbare Objekte der realen Welt oder unserer Vorstellung dar. Eine Entität wird durch Eigenschaften (Attribute) näher beschrieben.

Im vorliegenden Fahrschul-Projekt stellt beispielsweise der Fahrschüler „Andreas Abele“ eine Entität dar, die mit den Attributen 'nachname', 'vorname', 'telefon', 'email' usw. näher beschrieben wird.

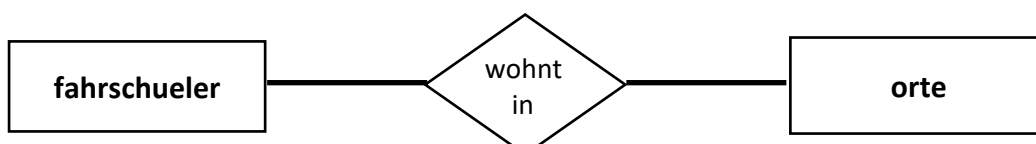
Die Entitäten, die durch dieselben Eigenschaften beschrieben werden, werden zu einem **Entitätstyp** zusammengefasst und in einem Rechteck dargestellt. Entitätstypen werden immer mit Substantiven benannt.

fahrschueler

Beziehungstyp

Wenn zwischen zwei oder mehreren Entitäten Zusammenhänge bestehen, werden diese durch Beziehungen zum Ausdruck gebracht. Gleichartige Beziehungen zwischen den Entitäten zweier Entitätstypen werden als Beziehungstyp bezeichnet und im ERD mit Hilfe einer Raute dargestellt.

Zwischen den Entitätstypen *Fahrschueler* und *Ort* besteht ein Beziehungstyp „wohnt in“. Im ERD wird dieser Zusammenhang folgendermaßen dargestellt:



Wenn die Fahrschule beispielsweise ihre Fahrschulautos und die Kfz-Händler, bei denen die Fahrzeuge gekauft wurden, in einer Datenbank erfassen will, ergeben sich im Datenmodell die Entitätstypen *autos* und *haendler*. Zwischen diesen beiden Entitätstypen besteht der Beziehungstyp „gekauft bei“.



Kardinalitäten

Die Kardinalität einer Beziehung definiert, wie viele Entitäten eines Entitätstyps mit genau einer Entität des anderen am Beziehungstyp beteiligten Entitätstyps (und umgekehrt) in Beziehung stehen können.

Die Kardinalität einer Beziehung wird mit folgender Frage bestimmt:

„Ein Exemplar des Entitätstyps 1 kann mit wie vielen Exemplaren des Entitätstyps 2 in Beziehung stehen?“

Konkret: **„Das Auto kann mit wie vielen Händlern in Beziehung stehen?“**

Antwort: „Das Auto kann mit **einem** Händler in Beziehung stehen.“

Für die Beziehung auto → haendler ergibt sich somit die Kardinalität 1.

Die Kardinalität der Beziehung haendler → autos ergibt sich aus der Frage:

„Der Haendler kann mit wie vielen Autos in Beziehung stehen?“

Antwort: „Der Haendler kann mit **vielen** Autos in Beziehung stehen.“

Für die Beziehung haendler → autos ergibt sich somit die Kardinalität N (N steht für die Antwort „viele“).

In der MySQL Workbench wird die N-Kardinalität mit dem Unendlichzeichen (∞) dargestellt.

Diese Art von Beziehung wird auch als 1 : N – Beziehung bezeichnet.

Im ERD wird dieser Zusammenhang folgendermaßen dargestellt:



Relationenmodell

Das Entity-Relationship-Diagramm wird verwendet, um die logischen Datenbankbeziehungen abzubilden. Bevor aus diesem Modell eine konkrete Datenbank implementiert werden kann, muss das ERD in einem Zwischenschritt in ein sogenanntes Relationenmodell (Relationenschema) überführt werden. Ziel dieser Umsetzung ist ein Modell, das möglichst direkt in einer konkreten relationalen Datenbank implementiert werden kann.

Für die im ersten Modellierungsschritt identifizierten Entitätstypen (*autos* und *haendler*) müssen in der zu erstellenden Datenbank jeweils Tabellen (Relationen) zur Verfügung gestellt werden. Dazu werden alle Entitätstypen in Relationen mit festgelegten Attributen überführt.

autos(autonr INT, kennzeichen VARCHAR, kaufpreis DOUBLE,)

haendler(haendlernr INT, firma VARCHAR, telefon VARCHAR, ...)

Die Überführung der Beziehungen erfolgt durch die Festlegung sogenannter Fremdschlüssel. Ein **Fremdschlüssel** kann Bestandteil einer Tabelle in einer relationalen Datenbank sein. Er wird benötigt, um Beziehungen zwischen Tabellen in der Datenbank zu realisieren. Dabei handelt es sich um ein Schlüsselattribut, das auf einen Primärschlüssel in einer anderen Tabelle verweist. Er stellt sicher, dass von einer Tabelle zu einer in Beziehung stehenden Tabelle navigiert werden kann.

Im vorliegenden Beispiel besteht zwischen den Entitätstypen *haendler* und *autos* eine 1 : N – Beziehung, die in der Datenbank zwischen den Tabellen *autos* und *haendler* realisiert werden muss.

Dazu wird das Primärschlüsselattribut *haendlernr* der Tabelle *haendler* (Parent-Tabelle) als Fremdschlüsselattribut *haendlernr* in die Tabelle *autos* (Child-Tabelle) übernommen.

So besteht die Möglichkeit, für jedes Auto die entsprechende Nummer des Händlers zu erfassen, bei dem es gekauft wurde. Die kompletten Daten eines Autos und seines Händlers können nun über die beiden Tabellen hinweg ermittelt werden.

Die Kennzeichnung eines Fremdschlüssels erfolgt im Relationenmodell mit einem vorangestellten senkrechten Pfeil (↑).

Relationenmodell:

autos(autonr INT, kennzeichen VARCHAR, kaufpreis DOUBLE, ..., ↑*haendlernr* INT)

haendler(haendlernr INT, firma VARCHAR, telefon VARCHAR, ...)

Ein Relationenmodell beschreibt also die in einer relationalen Datenbank zu implementierenden Relationen (Tabellen), deren Attribute, die festgelegten Datentypen sowie die verwendeten Schlüssel.